STORAGE NETWORK CONTROL METHOD

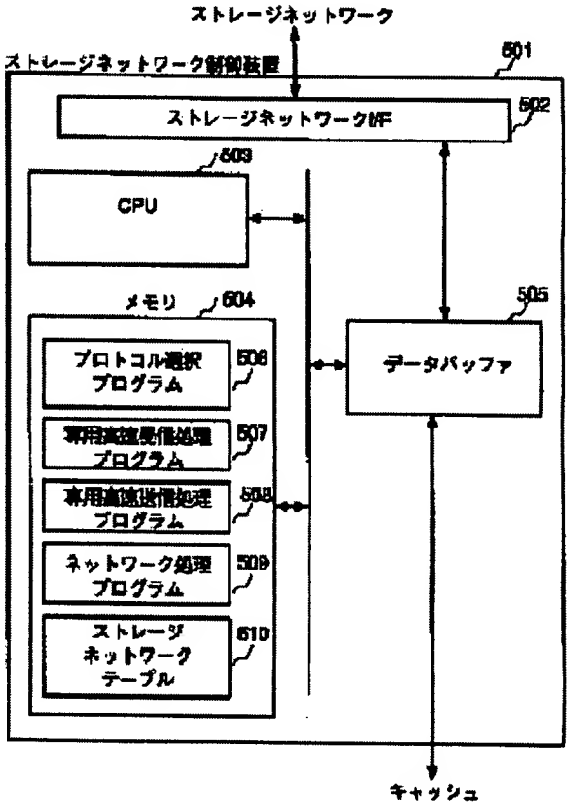
Abstract of Prior Art

Publication number: JP10069357
Publication date: 1998-03-10
Inventor: TAKAMOTO YOSHIFUMI; KANAI HIROKI
Applicant: HITACHI LTD
Classification:
- international: ***G06F3/06; G06F12/00; G06F13/00;*** **G06F3/06; G06F12/00; G06F13/00;** (IPC1-7): G06F3/06; G06F3/06
- European:
Application number: JP19960226401 19960828
Priority number(s): JP19960226401 19960828

Report a data error here

Abstract of **JP10069357**

PROBLEM TO BE SOLVED: To perform communication at a high speed between a disk array controllers while keeping a general purpose interface with a host by providing a mechanism for switching a communication protocol between the host and the disk array controller and the communication protocol between the disk array controllers.
SOLUTION: Information relating to devices connected on a storage network is stored in a storage network table 510. A protocol selection program 506 selects dedicated high-speed reception/transmission processing programs 507 and 508 or a network processing program 509 from the contents of the storage network table 510 and switches the protocol to be called corresponding to the result. Thus, the network processing program 509 is executed as before when a communication destination is the host and the dedicated high-speed reception/ transmission processing programs 507 and 508 are executed when it is the disk array controller.



Data supplied from the ***esp@cenet*** database - Worldwide

Prior Art 1

(19)日本国特許庁（ＪＰ）　(12) **公開特許公報**（Ａ）　(11)特許出願公開番号

特開平10－69357

(43)公開日　平成10年(1998)３月10日

(51)Int.Cl.°	識別記号	庁内整理番号	ＦＩ		技術表示箇所
Ｇ０６Ｆ　3/06	３０１		Ｇ０６Ｆ　3/06	３０１Ｐ	
				３０１Ｘ	
	５４０			５４０	

審査請求　未請求　請求項の数３　ＯＬ　（全 10 頁）

(21)出願番号　特願平8－226401

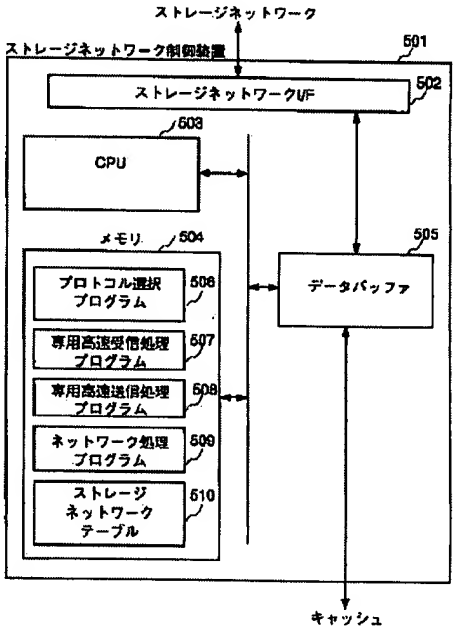
(22)出願日　平成８年(1996)８月28日

(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　高本　良史
東京都国分寺市東恋ケ窪一丁目280番地
株式会社日立製作所中央研究所内
(72)発明者　金井　宏樹
東京都国分寺市東恋ケ窪一丁目280番地
株式会社日立製作所中央研究所内
(74)代理人　弁理士　小川　勝男

(54)【発明の名称】　ストレージネットワーク制御方法

(57)【要約】
【課題】ディスクアレイ制御装置とホストプロセッサの通信プロトコルと、ディスクアレイ制御装置間の通信プロトコルを切り替える。
【解決手段】ディスクアレイ制御装置内にストレージネットワーク制御装置５０１を設け、この中に設けたストレージネットワークに接続された装置がホストプロセッサかディスクアレイ制御装置かを示すストレージネットワークテーブル５１０をサーチし、送信先に応じて通信プロトコルを切り替える。

図５



1

【特許請求の範囲】

【請求項１】複数のディスク装置を並列に動作させるとともに冗長データを格納する複数のディスクアレイ制御装置と、単一あるいは複数のホストプロセッサを接続するストレージネットワークを有するシステムにおいて、前記ストレージネットワークに接続された装置が、前記ホストプロセッサかディスクアレイ制御装置か、を示すストレージネットワークテーブルを前記ディスクアレイ制御装置内に設け、前記ディスクアレイ制御装置はデータの送信時に前記ストレージネットワークテーブルをサーチし、送信先が前記ホストプロセッサか前記ディスクアレイ制御装置かを判定し、前記ディスクアレイ制御装置であれば簡略化した通信方法を使用して通信を行うことを特徴とするストレージネットワーク制御方法。

【請求項２】請求項１において、前記ディスクアレイ制御装置および前記ホストプロセッサはデータ送信時に、送信元が前記ディスクアレイ制御装置か、前記ホストプロセッサかを示す識別子を送信データに付加し、前記ディスクアレイ制御装置は受信時に前記識別子を参照し、送信元が前記ディスクアレイ制御装置であれば前記簡略化した通信方法を使用するストレージネットワーク制御方法。

【請求項３】請求項１において、前記ストレージネットワークに接続された装置は、起動時に自装置識別子を送信し、前記自装置識別子を受信した装置は前記ストレージネットワークテーブルに追加するストレージネットワーク制御方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、ディスク制御装置に係り、ネットワークで相互に接続されたディスク制御装置のデータ通信制御方法に関する。

【０００２】

【従来の技術】一般にコンピュータシステムはプロセッサと二次記憶装置から構成されている。主として使用される二次記憶装置は磁気ディスク装置である。現在、ディスク記憶装置の容量の伸び率は極めて高いが、メカニカルな動作を伴う磁気ディスク装置の性能はプロセッサ性能の伸び率ほど高くない。その課題を解決する方法として、以下に示すディスクアレイが提案された。

【０００３】ディスクアレイは、磁気ディスクを並列に制御することで性能を向上させる方式である。D. Patterson, G. Gibson, and R. H. Kartzらによる“A Case forRedundant Arrays of Inexpensive Disks(RAID), in ACM SIGMOD Conference, Chicago, IL”, PP.109-116 (June 1988)(以下、第１の文献と呼ぶ)では、複数のディスクドライブにデータを分散して配置することでディスク内に格納されたデータへのアクセス時間を短縮し、かつパリティあるいはＥＣＣと呼ばれる冗長データを格納することで信頼性も高めるＲＡＩＤというディスクアレイの構成技術が紹介されている。つまり、アレイディスクでは、複数のディスクドライブに対して並列に入出力を行うので、データの読み出しあるいは書き込みは高速となり、また、ディスクドライブに障害が発生したときでもパリティと障害ディスクドライブ以外のデータから、障害ディスクドライブのデータを回復することができる。

2

【０００４】より具体的には、上記第１の文献では、データの格納方法によりＲＡＩＤレベルを複数に分類している。そのうち、製品で多く使用されるＲＡＩＤレベルは、ＲＡＩＤ１，ＲＡＩＤ３，ＲＡＩＤ５である。

【０００５】ＲＡＩＤ１はミラーリングであり、２台のディスクに同一データを格納することでディスクの障害に対する信頼性を高めている。読み出し時には、２台のディスクの内どちらか早いほうのディスクから読むことで単一ディスクに比べ高速である。ＲＡＩＤ３は、単一データをビットあるいはバイト毎にストライピング（分割）し、複数のディスクに並列に読み書きすることで、データ転送性能を向上可能である。ＲＡＩＤ５は入出力ブロックを単位として複数のディスクにストライピングを行うレベルである。

【０００６】ＲＡＩＤ３では単一入出力要求を小さな単位でストライピングするのに対し、ＲＡＩＤ５はそれよりも大きな単位でストライピングを行う。ＲＡＩＤ３は単一ユーザの大規模データ入出力を高速化することが主な目的であるが、ＲＡＩＤ５は多数ユーザの小規模データ入出力を並列に実行することが主な目的である。従って、ＲＡＩＤ３は特に大規模なデータ入出力が要求されるマルチメディアや科学技術計算に用いられる。ＲＡＩＤ５は多数ユーザのサービスを行うオンライン・トランザクションなどでのデータベース処理に用いられる。

【０００７】しかし、単一のディスクアレイでは性能や容量の増加に限界が生じる。そのために、ディスクアレイをさらに複数設け、並列度をより増やす方法がある。大容量で低コストのディスクアレイを構築する場合、単一のディスクアレイコントローラに多数のディスク装置を接続する構成が考えられるが、ディスクアレイ制御装置内の制御プロセッサの性能の限界のために容量は増加するが性能は伸びなくなってしまうという問題が発生する。

【０００８】制御プロセッサはホストからの入出力要求を解釈実行するために設けられる。制御しなければならないディスク装置の台数に比例した処理性能が要求されることから、現状では数台から数１０台規模のディスクアレイがほとんどである。

【０００９】この問題を解決する方法として、バスあるいは高速なネットワークで制御装置を接続することで拡張性を高める方法が考えられる。多数のディスクアレイを接続する方法は特開平7－44322号公報に述べられている。これは制御装置を階層化することで、各制御装置が

制御しなければならない台数を削減することが目的である。例えば、一つの制御装置で５台のディスク装置を制御できるとする。この制御装置を単一のバスあるいはネットワークに接続し、ホストプロセッサからはいずれの制御装置へもアクセスできるようにする。こうすることで全体で多数台のディスク装置を制御することが可能になる。

【００１０】

【発明が解決しようとする課題】ディスクアレイはホストプロセッサと接続され、ホストプロセッサからディスクアレイ制御装置に対して読み込みあるいは書き込みコマンドが発行されると、それに対応したデータの読み込みあるいは書き込みを行い、その実行結果をホストプロセッサに返送する。このような使用環境では、通信の形態は単一のホストプロセッサと複数のディスクアレイ制御装置との間で行われる。前述の特開平7－44322 号公報がその例である。

【００１１】しかし、ディスクアレイ制御装置間で自動的にバックアップの取得を行うような処理の場合、複数のディスクアレイ制御装置が接続されたバスあるいはネットワークを介して直接データを転送することになる。この場合問題となるのは通信方法（通信プロトコル）である。ディスク制御装置がホストプロセッサと通信を行う場合には、ホストプロセッサのインタフェースに合わせた通信プロトコルでデータの送受信を行う。しかし、ホストプロセッサのインタフェースは汎用である分、性能は低い。これは、ホストプロセッサが、ソフトウェアでデータのエラーチェックを行っていることが大きな要因である。

【００１２】ホストのインタフェースに合わせた通信プロトコルしか有さないディスクアレイ制御装置を使用して、前述のディスクアレイ制御装置間のバックアップを行った場合も同様の理由でバックアップ性能が低下することになる。これは、単一の通信プロトコルしか有していないことと、通信プロトコルを切り替えるための手段を有していないことが理由である。特開平7－ 44322 号公報では、複数のディスクアレイ制御装置を接続する形態が述べられているが、ディスクアレイ制御装置間で通信を行うことについては述べられていない。

【００１３】

【課題を解決するための手段】上記問題を解決するために、複数のディスク装置を並列に動作させるとともに冗長データを格納する複数のディスクアレイ制御装置と、単一あるいは複数のホストプロセッサを接続するストレージネットワークを有するシステムにおいて、前記ストレージネットワークに接続された装置が、前記ホストプロセッサかディスクアレイ制御装置か、を示すストレージネットワークテーブルを前記ディスクアレイ制御装置内に設け、前記ディスクアレイ制御装置はデータの送信時に前記ストレージネットワークテーブルをサーチし、送信先が前記ホストプロセッサか前記ディスクアレイ制御装置かを判定し、前記ディスクアレイ制御装置であれば簡略化した通信方法を使用して通信を行う。これにより、ホストプロセッサとの通信時は汎用インタフェースによる通信プロトコルを使用し、ディスクアレイ制御装置間で通信を行う場合には高速な通信プロトコルを使用することができる。

【００１４】

【発明の実施の形態】以下、本発明のストレージネットワーク制御方法を図面に示し実施例を参照してさらに詳細に説明する。

【００１５】図１は本発明によるストレージネットワーク制御方法を適用するディスク装置のブロック図である。１０１，１０２はホストであり、１０６，１０７はディスクアレイ制御装置であり，１０５はストレージネットワークである。ホストプロセッサ１０１，１０２はストレージネットワーク１０５を介してディスクアレイ制御装置１０６，１０７に相互接続されている。１３０，１３１はディスク装置であり、それぞれ複数のディスクドライブで構成されている。

【００１６】ホストプロセッサ１０１，１０２から発行されたディスクアレイ入出力要求は、ストレージネットワーク１０５を介してディスクアレイ制御装置１０６，107に転送され、またディスクアレイ入出力要求に伴うデータもストレージネットワーク１０５を介して転送される。従来から存在する主にコンピュータ間のデータ通信を主な目的とするネットワーク１３２は、ストレージネットワーク１０５とは独立に接続されている。ホストプロセッサ１０１，１０２内には、ストレージネットワーク１０５の制御を行うストレージネットワークインタフェース１０３，１０４が設けられる。

【００１７】ディスクアレイ制御装置１０６，１０７内は、ネットワークストレージ１０５を制御するインタフェース制御部１０６，１０７，ディスクアレイ制御装置106,１０７を制御するＣＰＵ１１０，１１１とマイクロプログラム１１８，１１９，ディスク装置１３０，１３１のデータを一時的に格納するキャッシュ１１２，１１３，ディスクコマンドを分配するスイッチ１１６，１１７，ディスク装置１３０，１３１との接続を制御するインタフェース制御１２０～１２４，１２５～１２９，パリティ制御１１４，１１５から構成されている。

【００１８】ディスクアレイ制御装置１０６，１０７の主な機能は、複数のディスクドライブを使用し、性能と信頼性を向上させるディスクアレイ制御を行うことである。ディスクアレイは、データやパリティの格納方法により、いくつかに分類されているが、本実施例では、それらのいずれも使用可能であり、また混在も可能である。それらの主な種類には前述の通り、ＲＡＩＤ1，ＲＡＩＤ3，ＲＡＩＤ5などがある。

【００１９】ＲＡＩＤ1，３，５ではいずれの場合も、

データ格納時に冗長データも同時にディスクに書き込むことで信頼性も向上している。具体的には、ＲＡＩＤ１の場合は、同一データを複数のディスクに書き込むことで冗長性を持たせている。またＲＡＩＤ３，５では、データを複数のディスクに分割するが、この時、旧データと旧パリティと新データとの排他的論理和を新パリティとしてディスクに書き込む。こうすることで、いずれかのディスクが障害を起こしても、ＲＡＩＤ１，３，５いずれの場合も障害を起こしたディスクのデータを回復することが可能である。

【００２０】例えば、ＲＡＩＤ３において以下のようにデータが格納されているとする。

ディスク１　　データ　＝１０１０１０１０
ディスク２　　データ　＝１１１１０００
ディスク３　　パリティ＝０１０１１０１０

この状態でディスク２が障害を起こし読み書きができなくなった場合、ディスク１とディスク３の排他的論理和を演算する。

ディスク１　データ　＝１０１０１０１０
ディスク３　パリティ＝０１０１１０１０
論理和　　＝１１１１０００〔ディスク２のデータ〕

このように、ディスク２のデータを再現することが可能である。図１のパリティ制御１１４，１１５はパリティの生成あるいはデータの回復を行う機構である。

【００２１】本発明は、ストレージネットワークの通信方法に関するが、図２を用いてストレージネットワークの特徴の一例を示す。ホストプロセッサ２０１，２０２はストレージネットワーク２０４を介してディスクアレイ制御装置２０５，２０６に相互接続されている。２０７，２０８はディスク装置であり、それぞれ複数のディスクドライブで構成されている。図２はディスクアレイ制御装置２０５からディスクアレイ制御装置２０６へデータのバックアップを行う動作を示している。ストレージネットワーク２０４を設けることで、ディスクアレイ制御装置２０５，２０６間で直接バックアップを取得することが可能となる（２０９）。

【００２２】従来のバックアップ〔図示せず〕は、ホストプロセッサがバックアップ元のディスクアレイ制御装置からバックアップデータの一部分を読み込み、そのデータをバックアップ先のディスクアレイ制御装置へネットワーク等を介して転送する操作を繰り返し行うことで処理していた。この従来の方法では、バックアップデータのすべてをホストプロセッサが処理しなくてはならず、バックアップ量が膨大になるとホストプロセッサはバックアップのために相当の能力を使用しなければならなくなる。しかし、ストレージネットワーク２０４を設けることで、ホストプロセッサはバックアップ要求２０３を一回発行するだけで、バックアップはディスクアレイ制御装置２０５，２０６間で自動的に取得することができ、またホストプロセッサ２０１，２０２へ処理負荷を与えることもなくなる。

【００２３】図３は本発明の概要を示した図である。ホストプロセッサ３０１，３０２はストレージネットワーク３０３を介してディスクアレイ制御装置３０６，３０７に相互接続されている。３１０，３１１はディスク装置であり、それぞれ複数のディスクドライブで構成されている。ストレージネットワーク３０３は、ディスクアレイ制御装置３０６，３０７とホストプロセッサ３１０，３０２間で通信すること以外に、ディスクアレイ制御装置３０６，３０７間でも通信可能である。

【００２４】本発明の特徴は、ホストプロセッサ３０１，３０２とディスクアレイ制御装置３０６，３０７間で通信する時と、ディスクアレイ制御装置３０６，３０７間で通信する時の通信プロトコルを自動的に切り替えることである。ホストプロセッサ３０１，３０２とディスクアレイ制御装置３０６，３０７間で通信する時にはネットワークプロトコルを使用し（３０４）、ディスクアレイ制御装置３０６，３０７間で通信する時は専用高速プロトコルを使用する（３０５）。

【００２５】ここで、プロトコルとは相互に情報を送受するための取り決められた通信方法である。通信におけるプロトコルは階層分けされている。一般には、物理層と論理層に分けられる。物理層は、データを運ぶためのケーブルの電気的特性等が決められている。また論理層では、ユーザデータを転送するためのソフトウェア的な通信手順が決められている。階層分けすることで、論理層が同一であれば物理層だけ異なるメディアを用いても通信可能となる。

【００２６】図３で示したネットワークプロトコルと専用高速プロトコルは論理層に対応している。ホストプロセッサ３０１，３０２を使用するユーザにストレージネットワークを意識させる必要がないようにするために、ユーザは従来のネットワーク１３２と同一の論理層のプロトコルを用いてストレージネットワークを使用することができる。これは、後で詳細に説明する。ユーザからは従来のネットワーク１３２もストレージネットワーク３０３もまったく同じ操作で使用可能である。これは、前述の論理層のプロトコルを従来のネットワーク１３２と同一のプロトコルに合わせたことの利点である。

【００２７】しかし、ネットワークプロトコル３０４は相互接続性では優れているが、性能面は高くない。これは、汎用性を持たせることと相反することであり、ハードウェア依存を少なくして通信できるようにするためにはハードウェアの機能を削減し、代わりにソフトウェアの機能を高くすることで実現されているためである。具体的には、ソフトウェアで転送時のエラーチェック機能を行っていることが性能低下の大きな原因になっている。これに対し、専用高速プロトコル３０５はエラーチェック機能を省略することで高速化が可能となってい

る。従来のネットワークとの互換はなくなるが、ユーザに見えない部分での通信であるためユーザへの影響はない。

【００２８】図４はホストプロセッサ４０１の構造を示している。ここでは、従来のネットワーク１３２とストレージネットワーク１０５がどのように共存しているかを示している。ホストプロセッサ４０１は、演算装置４０２，メモリ４０３，ネットワークインタフェース４０８，ストレージネットワークインタフェース４０９から構成されている。

【００２９】演算装置４０２はメモリ４０３から命令を読み取り解釈／実行を行う。メモリ４０３内には、ユーザプログラム４０４，ネットワーク処理プログラム４０５，ネットワークドライバ４０６，ストレージネットワークドライバ４０７が格納されている。ユーザプログラム４０４は、ネットワーク処理プログラムに対して入出力要求を発行すると、要求内容によってネットワークドライバ４０６あるいはストレージネットワークドライバ４０７に処理を切り替えて要求を発行する。それを受けたネットワークドライバ４０６あるいはストレージネットワークドライバ４０７は、異なるハードウェアであるネットワークインタフェース４０８とストレージネットワークインタフェース４０７をそれぞれ制御し、接続相手と通信を行う。ネットワーク処理プログラムは図３で述べた、論理層のプロトコルを処理する。ユーザは、通信相手がネットワークかストレージネットワークかを意識することなくプログラムを開発し使用することが可能となる。

【００３０】図５は本発明の特徴であるストレージネットワーク制御を行う制御装置の構成を示している。ストレージネットワーク制御装置５０１はストレージネットワークインタフェース５０２，ＣＰＵ５０３，メモリ５０４，データバッファ５０５から構成されている。ストレージネットワークインタフェース５０２はストレージネットワーク１０５に接続するための電気的な制御を行う。ＣＰＵ５０３はストレージネットワーク制御装置５０１の全体を制御する。具体的には、メモリ５０４に格納されているプログラムを随時読み出し、解釈／実行を行う。メモリ５０４内には、プロトコル選択プログラム５０６，専用高速受信処理プログラム５０７，専用高速送信プログラム５０８，ネットワーク処理プログラム５０９，ストレージネットワークテーブル５１０が格納されている。

【００３１】プロトコル選択プログラム５０６は、専用高速受信／送信処理プログラム507，５０８かネットワーク処理プログラム５０９かのいずれかを選択し、呼び出す処理を行う。この処理については後で詳細に説明する。専用高速送信／受信処理プログラム５０７，５０８はネットワーク処理プログラム５０９と比較し汎用性はないが、処理が高速であることが特徴である。この処理については、後で詳細に説明する。

【００３２】ストレージネットワークテーブル５１０はストレージネットワーク上に接続された装置に関する情報が格納されている。プロトコル選択プログラム５０６は、ストレージネットワークテーブル５１０の内容から、どの処理プログラムを選択するかを決定する。データバッファは、キャッシュあるいはストレージネットワークから転送されたデータを一時的に格納するために使用される。また、専用高速送信／受信処理プログラム５０７，５０８やネットワーク処理プログラム509がデータにヘッダを付加するためにも使用される。

【００３３】図６，図７はストレージネットワークに接続された装置の例と、その場合のストレージネットワークテーブル５１０の例を示している。

【００３４】図６では、ストレージネットワークに２台のホスト６０１，６０２と２台のディスクアレイ制御装置６０４，６０５が接続されている。２台のホスト６０１，６０２と２台のディスクアレイ制御装置６０４，６０５はストレージネットワーク６０３で相互通信が可能な構成で接続されている。２台のホスト６０１，602は、２台のディスクアレイ制御装置６０４，６０５のいずれからもデータ入出力が可能である。また、ディスクアレイ制御装置６０４，６０５間でもデータ送受信が可能である。ホスト６０１，６０２は、それぞれネットワークストレージ識別子１，２が定義されている。また、ディスクアレイ制御装置６０４，６０５にも同様にネットワークストレージ識別子３，４が定義されている。

【００３５】図７はこの構成例における、ストレージネットワークテーブル５１０内の構造を示している。テーブルのカラム７０１はストレージネットワーク識別子が格納されている。テーブルのカラム７０２はストレージネットワーク識別子に対応した装置の種類が格納されている。例えば、ホストであれば０が、ディスクアレイ制御装置であれば１が格納されている。プロトコル選択プログラム５０６は、このテーブルの内容から、要求されたコマンドやデータをどのような装置に転送するのかサーチし、その結果に応じて呼び出すべきプロトコルを切り替える。具体的には、通信先がホストであれば従来どおりネットワーク処理プログラムを実行し、ディスクアレイ制御装置であれば専用高速送信／受信処理プログラムを実行する。

【００３６】ストレージネットワークテーブル５１０の生成は、システム起動時にストレージネットワークに接続された装置間で情報の受け渡しを行うことで実現可能である。具体的には、自システム起動時にストレージネットワークに接続された全ての装置に対し自装置の識別子を発行する。識別子の転送は識別子転送用のコマンドを使用する。このコマンドを受けた装置は、自装置内のストレージネットワークテーブルに送信された識別子を追加する。

【００３７】図８はプロトコル選択プログラムの処理フローを示している。

【００３８】ステップ８０１では要求が送信要求か受信要求かを判断する。送信要求であればステップ８０２に移り、受信要求であればステップ８０７に移る。ステップ８０２では送信先識別子を取得する。これはキャッシュ１０８，１０９からデータバッファ５０５内に転送されたデータにデータと共に格納されている。この詳細は後で説明する。ステップ８０３では、図７で説明したストレージネットワークテーブルを送信先識別子をキーとしてサーチする。

【００３９】ステップ８０４ではステップ８０３でサーチした結果を参照し、送信先がホストであればステップ８０５に移り、送信先がディスクアレイ制御装置であればステップ８０６に移る。ステップ８０５では、ホスト内のネットワーク処理プログラム４０５と同様の処理を行う。ステップ８０６では、専用高速送信処理プログラムの呼び出しを行う。この処理は後で詳細フローを示す。ステップ８０７は受信処理時に実行され、データに付加された情報からプロトコルを取得する。本実施例では、ホストから転送された場合は０が、ディスクアレイから転送された場合には１が格納されている。データ構造は後で詳細に説明する。

【００４０】ステップ８０８では、送信元がホストかどうか判断する。この判断では、ステップ８０７で取得したプロトコルから判断可能である。ホストから転送された場合にはステップ８０９に移り、ディスクアレイ制御装置から転送された場合にはステップ８１０に移る。ステップ８０９では、ホスト内のネットワーク処理プログラム４０５と同様の処理を行う。ステップ８１０では、専用高速受信処理プログラムの呼び出しを行う。この処理は後で詳細フローを示す。このプロトコル選択プログラムにより、汎用インタフェースを要求する場合には、従来のネットワーク処理を行い、性能を要求する場合には専用高速送信／受信に切り替えることができる。

【００４１】図９はネットワーク処理に使用されるデータ形式を示している。本データ構造は、ホストとディスクアレイ制御装置との間で通信を行う際に、ストレージネットワークを流れるデータの構造を示している。ホストからディスクアレイ制御装置に転送されたデータや、ディスクアレイ制御装置がホストに対してデータを送信する場合には図９のようなデータ形式で通信を行う。

【００４２】９０１はストレージネットワークヘッダであり、ストレージネットワーク専用の制御情報が先頭に付加される。９０７はネットワークヘッダであり、ホスト内のネットワーク処理プログラム４０５が認識可能なデータのヘッダが付加される。９１２はユーザのコマンドあるいはデータである。ネットワークでは、図９に示すように階層を渡る毎にヘッダが付加される。送信側では、ネットワーク処理プログラムがまずネットワークヘッダ９０７を付加し、ストレージネットワークを通過する際ストレージネットワークヘッダ９０１を付加する。受信側では、まずストレージネットワークヘッダ９０１が取り除かれ、次にネットワーク処理プログラムでネットワークヘッダ９０７が取り除かれ、ユーザプログラムに渡される。これはカプセル化と呼ばれており、各ヘッダを付加するために処理オーバヘッドがかかるが、ユーザプログラムを変更する必要がないことから一般的に採用されている方法である。

【００４３】ネットワークヘッダ９０７内の９０８はユーザデータ長、９０９は送り元ネットワーク識別子、９１０は送り元ネットワーク識別子、９１１はエラーチェックコードが格納されている。９０８から９１０はデータを処理する上で必要な情報であるが、エラーチェックコード９１１はデータが正しく転送されたかどうかをチェックするために送信側でコードが計算され格納される。受信側では、送信側と同じようにエラーチェックコードを生成し転送されたエラーチェックコードと比較する。その結果、値が等しければデータは正しく転送されたと判断し、値が異なっていればデータの再送を送信側に要求する。

【００４４】エラーチェックコードにはいくつかの種類があるが、一般的に用いられるのはチェックサムと呼ばれる方法で、送信するデータの内容を数バイト単位に区切り、和をとる方法である。この方法は、エラーチェック方法の中でも処理が簡単ではあるが、エラーチェックコードを生成するのにユーザデータ長に比例した時間がかかるため大きなオーバヘッドとなる。

【００４５】上記と同様にストレージネットワークヘッダ９０１内の９０２はプロトコル識別子、９０３はネットワークヘッダ９０７とユーザデータ＆コマンド９１２の長さの合計、９０４は送り元識別子、９０５は送り先識別子、９０６はエラーチェックコードが格納されている。プロトコル識別子９０２は、ストレージネットワークヘッダ９０１の後に続くデータがどのようなプロトコルで転送されたかを示す識別子で、ネットワーク処理に使用される場合には０が格納される。エラーチェックコード９０６は、ネットワークヘッダ９０７で述べたエラーチェックコード９１１と同様である。機能としてはネットワークヘッダ９０７のエラーチェックと同様であり、２重のエラーチェックで信頼性を向上させる効果はあるが、処理の上ではオーバヘッドが大きくなり転送性能は低下する。従来のネットワークもストレージネットワークも共に規格化されたプロトコルであり、またユーザインタフェースを変えない場合にこのようなオーバヘッドが生じる。

【００４６】図１０は専用高速送信／受信の場合のデータ構造を示している。１００１はストレージネットワークヘッダであり、１００７はユーザデータ＆コマンドである。ストレージネットワークヘッダ１００１内の１０

０２はプロトコル識別子、１００３はユーザデータ＆コマンド１００７の長さの合計、１００４は送り元識別子、１００５は送り先識別子、１００６はエラーチェックコードが格納されている。

【００４７】プロトコル識別子１００２は、ストレージネットワークヘッダ１００１の後に続くデータがどのようなプロトコルで転送されたかを示す識別子で、専用高速送信／受信の場合は１が格納される。これは受信したデータが、ホストから転送されたのかディスクアレイ制御装置から転送されたのかを判断するのに用いられる。専用高速送信／受信の場合は、図９の場合と異なりネットワークヘッダ９０７がない。ネットワーク形式のデータ構造ではないため、この形式ではホストと通信することはできないが、エラーチェックコードを２重に生成する必要がないため処理を高速化できる。

【００４８】図１１は、キャッシュ１１２，１１３からデータバッファ５０５に転送されたデータの構造を示している。１１０１はキャッシュヘッダ、１１０４はユーザデータ＆コマンドが格納されている。このデータはマイクロプログラム１１８，１１９により生成される。キャッシュヘッダ１１０１内の１１０２は送り先識別子、１１０３はオプションにより付加されることもされないこともあるが、通常データ長等が格納されている。送り先識別子１１０２を参照することで、プロトコル選択プログラム（図８）内のステップ８０３で、どこにデータやコマンドを送信するのかを判断することができる。

【００４９】図１２は専用高速受信プログラムの処理フローを示している。ステップ1201ではエラーチェックコードの確認を行う。これは、受信したデータのエラーチェックコードを生成し、図１０のエラーチェックコード１００６と比較することで実現可能である。ステップ１２０２ではエラーが発生したかどうかを判断する。エラーが発生していた場合、ステップ１２０７に移り、送信元に対し再送を要求し処理を終了する。エラーがなかった場合、ステップ１２０３に移る。ステップ１２０３では、送り元識別子を取り出す。ステップ１２０４では、ストレージネットワークヘッダを削除し、代わりにネットワークヘッダを付加する。ステップ１２０５では、キャッシュヘッダ内の送り先識別子１１０２にステップ１２０３で取り出した送り先識別子を格納する。これは処理終了後、ユーザへデータやコマンドを返送する際に使用される。ステップ１２０６では、ステップ１２０５で生成したデータをキャッシュへ転送する。ディスクアレイ制御装置は、キャッシュにデータが転送されると転送された内容を解釈し実行する。

【００５０】このように専用高速受信は、ネットワーク処理と異なり、エラーチェックコードチェック回数を削減することができるため処理を高速化できる。

【００５１】図１３は専用高速送信処理プログラムの処理フローを示している。ステップ１３０１では、キャッシュヘッダから送り先識別子の取り出しを行う。ステップ１３０２では、ストレージネットワークヘッダ内のプロトコル識別子を１に設定する。これは、専用高速送信データであることを受信側に知らせる意味を持つ。ステップ１３０３では、ストレージネットワークヘッダにデータ長を格納する。ステップ１３０４では、ストレージネットワークヘッダに送り先識別子を格納する。これはステップ１３０１でキャッシュヘッダから取得した識別子である。ステップ１３０５では、送り元識別子を格納する。ステップ１３０６ではエラーチェックコードを生成し、ストレージネットワークヘッダに格納する。このように専用高速送信は、ネットワーク処理と異なり、エラーチェックコードチェック回数を削減することができるため処理を高速化できる。

【００５２】上記処理により、ディスクアレイ制御装置間でデータを高速に送受信することができるようになる。また、ユーザはこれらの切り替えは意識する必要がない。

【００５３】一般にネットワークは、バスあるいはループのトポロジの意味が強いが、本発明はトポロジには依存しない。そのため、スイッチを使用したスター形式のトポロジでも発明の効果は得られる。

【００５４】

【発明の効果】本発明では、複数のディスクアレイ制御装置をネットワークで接続した形態のストレージシステムにおいて、ホストとディスクアレイ制御装置間における通信プロトコルと、ディスクアレイ制御装置間における通信プロトコルを切り替える機構を設けることにより、ホストとは汎用インタフェースを保ちつつ、ディスクアレイ制御装置間は高速に通信を行うことができる。

【図面の簡単な説明】

【図１】本発明における実施例の全体のブロック図。

【図２】本発明の実施例におけるネットワークストレージのブロック図。

【図３】本発明の実施例における動作概要を示すブロック図。

【図４】本発明の実施例におけるホストプロセッサのブロック図。

【図５】本発明の実施例におけるストレージネットワーク制御装置のブロック図。

【図６】本発明の実施例におけるストレージネットワークのブロック図。

【図７】本発明の実施例におけるストレージネットワークテーブルを示す説明図。

【図８】本発明の実施例におけるプロトコル選択プログラムの処理のフローチャート。

【図９】本発明の実施例におけるネットワークプロトコル転送時のデータ構造を示す説明図。

【図１０】本発明の実施例における専用高速送信／受信プロトコル転送時のデータ構造を示す説明図。

13

【図１１】本発明の実施例におけるキャッシュヘッダの構造を示す説明図。

【図１２】本発明の実施例における専用高速受信プログラムのフローチャート。

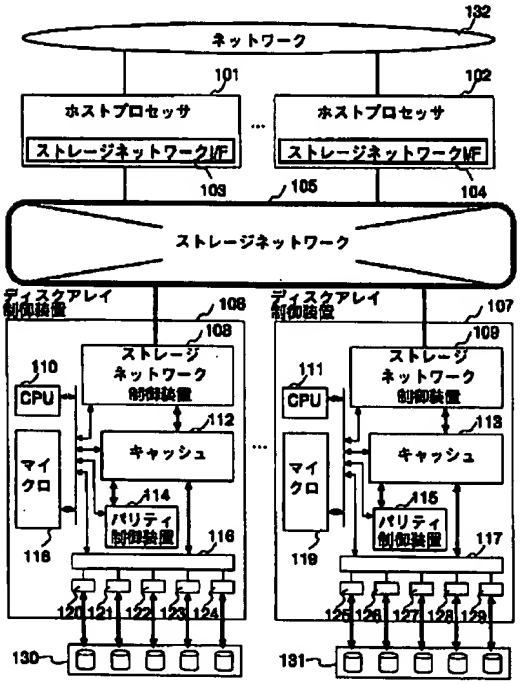
【図１３】本発明の実施例における専用高速送信プログラムのフローチャート。

【符号の説明】

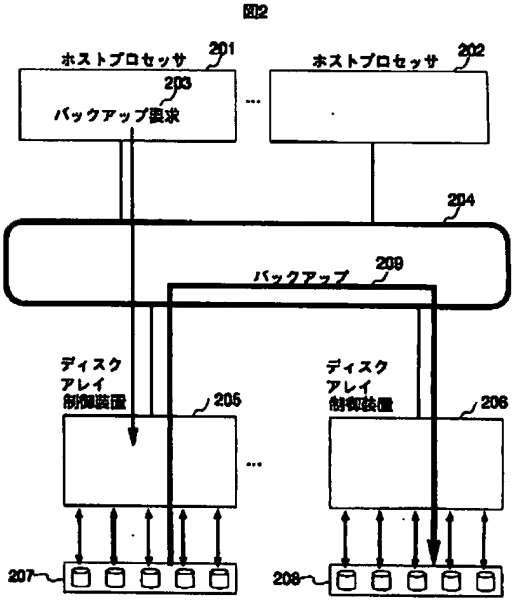
14

＊１０１，１０２…ホストプロセッサ、１０６，１０７…ディスクアレイ制御装置、１０５…ストレージネットワーク、１０８，１０９…ストレージネットワーク制御装置、５０６…プロトコル選択プログラム、５０７…専用高速受信処理プログラム、５０８…専用高速送信処理プログラム、５１０…ストレージネットワークテーブル。

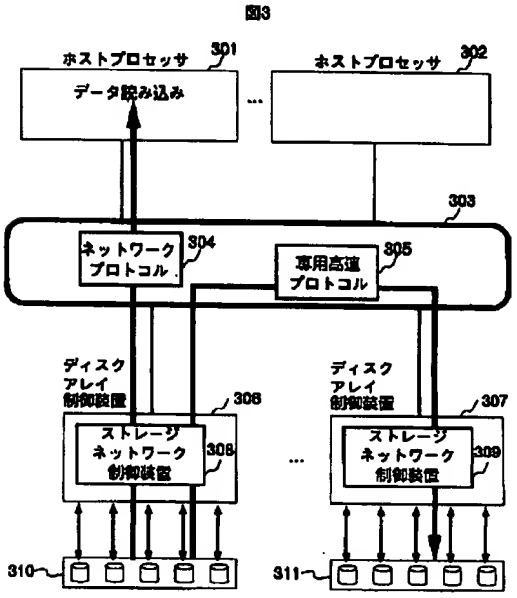
【図１】



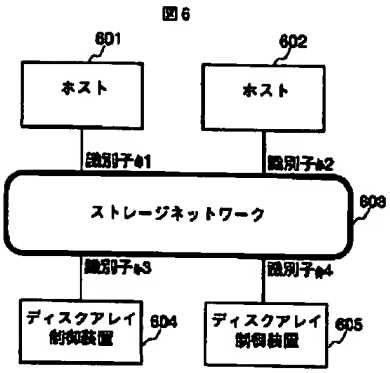
【図２】



【図３】

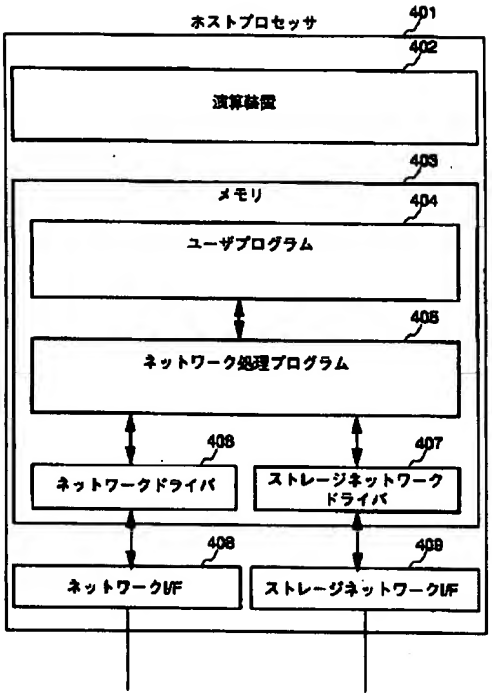


【図６】



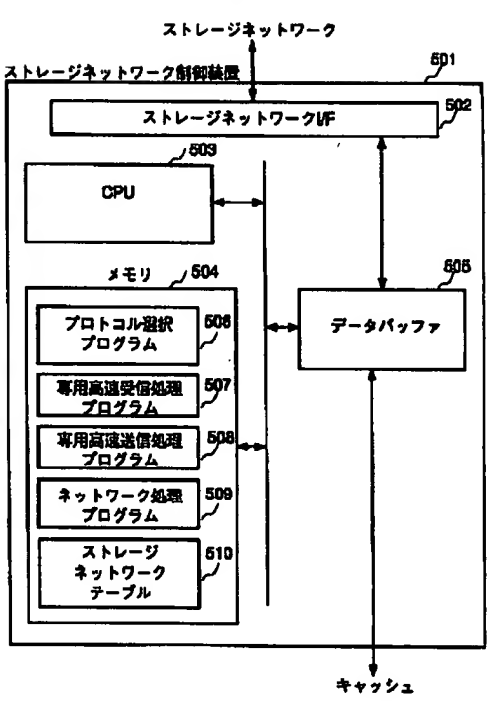
【図4】

図4



【図5】

図5



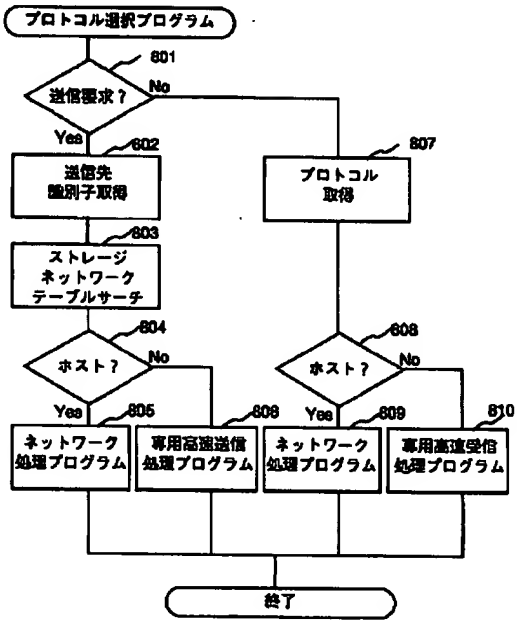
【図7】

図7

701 ストレージネットワーク識別子	702 装置種別
1	0 (ホスト)
2	0 (ホスト)
3	1 (制御装置)
4	1 (制御装置)

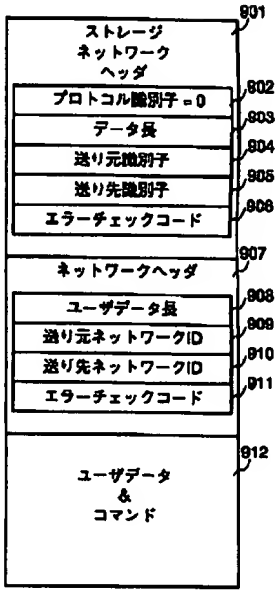
【図8】

図8



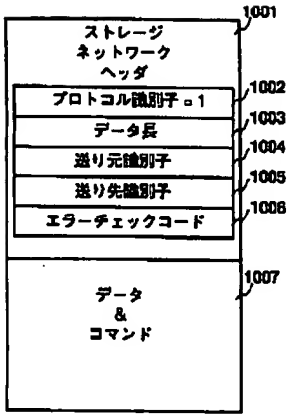
【図9】

図9



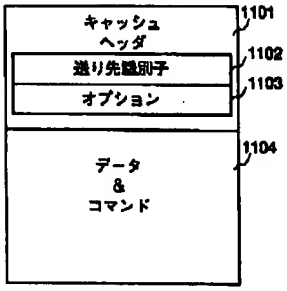
【図１０】

図１０



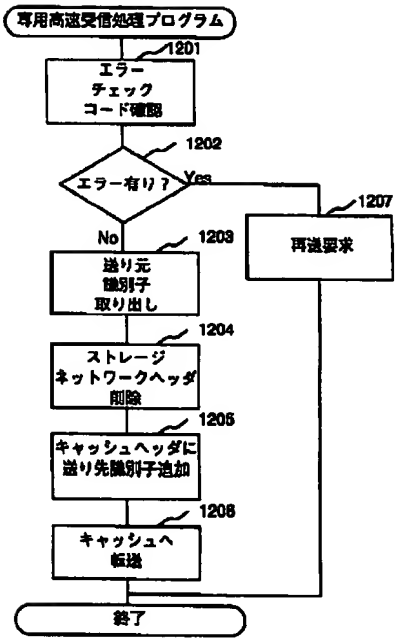
【図１１】

図１１



【図１２】

図１２



【図１３】

図１３